

Object-Oriented Programming Language – Midterm
(Part II, open textbook, notes, compiler, 150 minutes, 試卷隨答案卷繳回)

10/24/2013

Name: _____

ID: _____

Download FirstMidterm.zip **from the course website.**

1. (20%) Read the words from oldMan.txt given in FirstMidterm.zip. After you've read all the words, process them and change the first letter of each word to uppercase. Print the transformed elements, five words to a line. Your output looks like:

```
Enter the file name: noSuchFile.txt
Complain: I cannot find the file
磁碟區 C 中的磁碟沒有標籤。
磁碟區序號: 3690-9459

C:\Users\IAN\Documents\Visual Studio 2013\Projects\Mock_Prob1\Mock_Prob1 的目錄

2014/10/14 上午 05:51 <DIR> .
2014/10/14 上午 05:51 <DIR> ..
2014/10/14 上午 05:51 <DIR> Debug
2014/10/14 上午 05:38 3,449 Mock_Prob1.vcxproj
2014/10/14 上午 05:38 1,067 Mock_Prob1.vcxproj.filters
2014/10/14 上午 05:38 260 oldMan.txt
2014/10/14 上午 05:51 1,037 Source.cpp
4 個檔案 5,813 位元組
3 個目錄 382,413,123,584 位元組可用

Enter the file name: oldMan.txt
The lines with transform elements are:

The Old Man And The
Sea Is One Of Hemingway's
Most Enduring Works.
Here Hemingway Recasts, In Strikingly
Contemporary Style, The Classic Theme
Of Courage In The Face
Of Defeat, Of Personal Triumph
Mon From Loss.
The Old Man And The.
The Old Man And The
Sea.
請按任意鍵繼續 . . .
```

Notice if your original file has an exact 5 words in a line, you should not add a new line after it (highlighted above in the sample run).

2. (20%) (a) Write an IntegerVector class for an array of integers. In the class, use std::vector as your internal data representation and provide the following constructors:

Object-Oriented Programming Language – Midterm

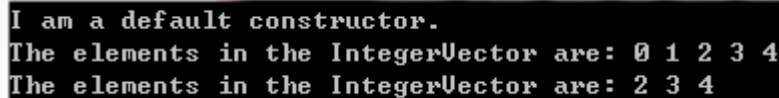
(Part II, open textbook, notes, compiler, 150 minutes, 試卷隨答案卷繳回)

- `IntegerVector()` // print a message said “I am a default constructor.”
- `IntegerVector(unsigned nelems)` // creates an `IntegerVector` with the integers `0...nelems-1`
- `IntegerVector(unsigned start, unsigned end)` // creates an `IntegerVector` with the range `[start, end)`

In addition, provide a non-member `print` function. Test your program with the client code `Prob2a.cpp` listed below and given in the `FirstMidterm.zip`:

```
#include <iostream>
#include "IntegerVector.h"
using namespace std;

int main()
{
    IntegerVector a;
    IntegerVector b(5);
    print(cout, b);
    IntegerVector c(2, 5);
    print(cout, c);
}
```



I am a default constructor.
The elements in the IntegerVector are: 0 1 2 3 4
The elements in the IntegerVector are: 2 3 4

(b) Add an intersection member function that prints out the elements common to two arrays. Allow a sequential operation. Test your program with the client code `Prob2b.cpp` listed below and given in the `FirstMidterm.zip`:

```
#include <iostream>
#include "IntegerVector.h"
using namespace std;

int main()
{
    IntegerVector a(3);
    IntegerVector b(5);
    IntegerVector c(2, 6);
    print(cout, c.intersection(b).intersection(a));
}
```

Your output looks like:

Object-Oriented Programming Language – Midterm
(Part II, open textbook, notes, compiler, 150 minutes, 試卷隨答案卷繳回)

```
Intersection elements are: 2 3 4
Intersection elements are: 2
The elements in the IntegerVector are: 2
```

3. (20%) C++ does not provide a built-in way of representing or manipulating the mathematical polynomial such as $6x^3-5x+10.2$. One way to represent the polynomial is to use `vector<double>` to store the coefficients in which the index of the vector is the exponents of the corresponding terms and the element of the vector is the coefficient. A home-made input syntax via string allows us to parse the input. For example, a very simple home-made string representation of $6x^3-5x+10.2$ is:

```
| 10.2 x^0 -5 x^1 0 x^2 6 x^3
```

And the contents of the `vector<double>` for the polynomial representation are:

```
| [10.2, -5.0, 0.0, 6.0]
```

Write a Polynomial class that provides a constructor and add and print member functions. Put your class definition in a header file `poly.h` and implement the constructor and member functions in a cpp file `poly.cpp`. Test your program with the client code `Prob3.cpp` listed below and given in the `FirstMidterm.zip`:

```
#include "Polynomial.h"
#include <string>
#include <iostream>
using namespace std;

int main ()
{
    cout << "Please enter the first polynomial with a home-made
string representation: " << endl ;
    string poly_rep ;
    getline(cin, poly_rep) ;
    Polynomial poly1(poly_rep) ;

    cout << "Please enter the second polynomial with a home-made
string representation: " << endl ;
    getline(cin, poly_rep) ;
    Polynomial poly2(poly_rep) ;

    cout << "The first polynomial after the addition of the second
polynomial is: " << endl;
    poly1.combine(poly2).print(cout) << endl ;

    return 0 ;
}
```

Object-Oriented Programming Language – Midterm
(Part II, open textbook, notes, compiler, 150 minutes, 試卷隨答案卷繳回)

```
| }
```

Two sample runs look like:

```
Please enter the first polynomial with a home-made string representation :
1.2 x^0 3 x^1 0 x^2 3.5 x^3
Please enter the second polynomial with a home-made string representation :
0.3 x^0 -2.2 x^1 1.2 x^2
The first polynomial after the addition of the second polynomial is(print) :
1.5 x^0 0.8 x^1 1.2 x^2 3.5 x^3
請按任意鍵繼續 . . .
```

```
Please enter the first polynomial with a home-made string representation :
1.2 x^0 2.3 x^1
Please enter the second polynomial with a home-made string representation :
3 x^0 2.1 x^1 1.2 x^2 0 x^3 3.2 x^4
The first polynomial after the addition of the second polynomial is(print) :
4.2 x^0 4.4 x^1 1.2 x^2 0 x^3 3.2 x^4
請按任意鍵繼續 . . .
```

4. (10%, you will need to finish Problem 3 before doing Problem 4) Now release the format constraint in Problem 3 so you can parse a polynomial $6x^3-5x+10.2$ represented by:

```
| 6 x^3 -5 x^1 10.2 x^0
```

Or

```
| -5 x^1 10.2 x^0 6 x^3
```

Test your implementation using the same client code given in Problem 3 (Prob3.cpp) but enter the first and second polynomials that support the new polynomial representations. Report the first polynomial with the same client codes used in Problem 3. Below is a sample run.

```
Please enter the first polynomial with a home-made string representation :
6 x^3 -5 x^1 10.2 x^0
Please enter the second polynomial with a home-made string representation :
-5 x^1 10.2 x^0 6 x^3
The first polynomial after the addition of the second polynomial is(print) :
20.4 x^0 -10 x^1 0 x^2 12 x^3
請按任意鍵繼續 . . .
```